Managing Heterogeneous Models and Schemas in the Waterway Information Network

R. M. Malyankar[1], K. M. Shea[2], J. W. Spalding[2], M. J. Lewandowski[3], A. R. Baddam[1]

[1] Dept, of Computer Science
Arizona State University
Tempe, AZ 85287.
rmm@acm.org,anvith@asu.edu

[2] U. S. Coast Guard Research and Development Center
Groton, CT 06340.
{KShea,JSpalding}@rdc.uscg.mil

[3] Potomac Management Group
214 Thames Street
Groton, CT 06340.
mlewandowski@rdc.uscg.mil

Project URL: http://www.eas.asu.edu/~gcss/research/navigation/

Abstract

Managing models and schemas pertaining to different parts of the marine information domain is an important part of the U.S. Coast Guard R&D Center's proposed Waterway Information Network. This paper describes our approach to distributed model and schema management and an architecture for facilitating the use of formal artificial intelligence concepts in application development with XML.

1. Introduction

The Waterway Information Network (WIN) proposed by the United States Coast Guard (USCG) R&D Center is intended to make information transfer in the Marine Transportation System (MTS) more efficient, accurate, and timely. WIN will use a distributed content management architecture and a tailored XML-based markup language called MIML (Maritime Information Markup Language). Many MTS stakeholders already have their own information models (constructed independently over several years), and creating and managing a new single information model or XML schema covering all the diverse sources of data would require a large investment of time and resources and be an extremely complex task for logistical and technical reasons. An approach to integrating and managing different kinds of models and schemas is needed. This paper describes such an approach. The significant issues addressed by the architecture described in this paper are: (1) managing diverse models derived from different sources; (2) providing different schemas for different application areas while maintaining application interoperability; and, (3) making information distribution easier for information suppliers and consumers, in part by moving from paper-based means to electronic means of distribution.

2. The Waterway Information Network

The MTS has an extremely diverse community: government entities and agencies (federal, state, and local); military and non-military users; and many commercial, private, and recreational members. Some entities derive profit from supplying information or by adding value to public information. Information transfer arrangements currently consist of a multitude of information “stovepipes”, whereby providers transfer a single type of information through one or more methods to their user comunities in specific formats. Examples are Local Notices to Mariners, marine weather information broadcasts and the Physical Oceanographic Real-Time System; in these cases, multiple government agencies provide information to a wide range of users, including elements of the government agencies themselves. Other information flows exist: some information is transferred solely to various government agencies; a ship’s agent notifies at least four federal agencies about a vessel’s arrival and must also notify or arrange services with tugs, pilots, stevedores, terminalling, chandlering, etc. Similar information is thus passed multiple times to multiple information users. Also, many users are also producers, particularly waterway users, who send updates about navigation aid status, dangers, etc., to the original providers.

Shortcomings in current methods of information transfer in the MTS include a reliance on paper-based systems; individual and distinct methods and procedures for submitting and disseminating information; and numerous marine electronic information devices and systems that are not part of a fully integrated system. Studies indicate a need for interoperable, Internet-enabled information resources that are up-to-date, accurate, non-redundant, easily accessible, available in multiple formats, and decentralized (USDoT 1999, NRC 1999, USCG-PMG 2001).

WIN will provide an integrated Internet-based solution as an alternative to the "stovepipe" information transfer process; emphasize distributed content management where information providers retain control of their data; and facilitate bi-directional flow of information, all without a central hub. Given the diversity and number of information providers, WIN must include different models and schemas corresponding to different interests and sub-domains of marine information, and provide a mechanism for distributed management of models and schemas. WIN will also provide an XML markup language, MIML, for information exchange. The next section describes our technical solution to the problem of distributed definition and management of models and schemas (i.e., the markup vocabulary) for WIN/MIML. (This is complemented by a "process" solution, omitted for brevity.)

3. Model and Schema Management

A model describes a knowledge domain in terms of entities and their relationships. We use the term *model* to cover computational ontologies as well as less formal models, e.g., tabulations of data items and relations. An ontology is a formal representation of a domain in terms of classes, sub-classes, instances, relationships, constraints, etc., and is based on a formal representation language with rigorous and well-defined semantics. The advantage of such a formal representation is that its semantics are the same in any application, which makes the representation reusable across applications, and that general-purpose software can be used for searching, inferencing, etc. Ontologies are increasingly being used to support intelligent searching, knowledge-base navigation, inferencing and reasoning. We hope to use ontological representations in WIN where possible, so as to allow future intelligent applications such as reasoning and hierarchical indexing of knowledge bases, and to provide applications with clear and unambiguous semantics for entities and relationships. Given that application developers will need XML schemas, it is necessary to provide a way of managing different models and schemas and making any linkages between models and schemas clear to modelers, schema designers, and application developers.

Our approach to the three issues mentioned in Section 1 is based on allowing models to be expressed in any of a selected set of standard forms and providing convertors to generate XML schemas directly from the models. These initial schemas are *type libraries* in that their contents describe entities and attributes taken from the models, instead of document elements – they describe the data rather than the documents. Other schemas are derived from these type libraries; these derived schemas describe the documents used in the domain. Figure 1 shows the relationships between examples of models, type library schemas, and application schemas. (Model names in the figure are illustrative examples.) One view of the different domains (and hence the differenr subdivisions of MIML) is given in (Malyankar 2002), which also discusses the relationship between ontologies and markup vocabulary.

The process of creating type libraries is being automated using a schema generator (one generator for each model format). The type libraries are application independent and are used to derive application schemas in the next level in the schema hierarchy. These schemas will be created and maintained by model and schema designers. Both type libraries and application schemas can be used by application schema designers to create other application schemas. The application schemas can be directly used by various applications, e.g., database update, vessel tracking, etc. Figure 1 shows examples of application schemas, such as a "C.P. Schema" for Coast Pilot-type information and an "ENC Update Schema" for distributing chart updates.

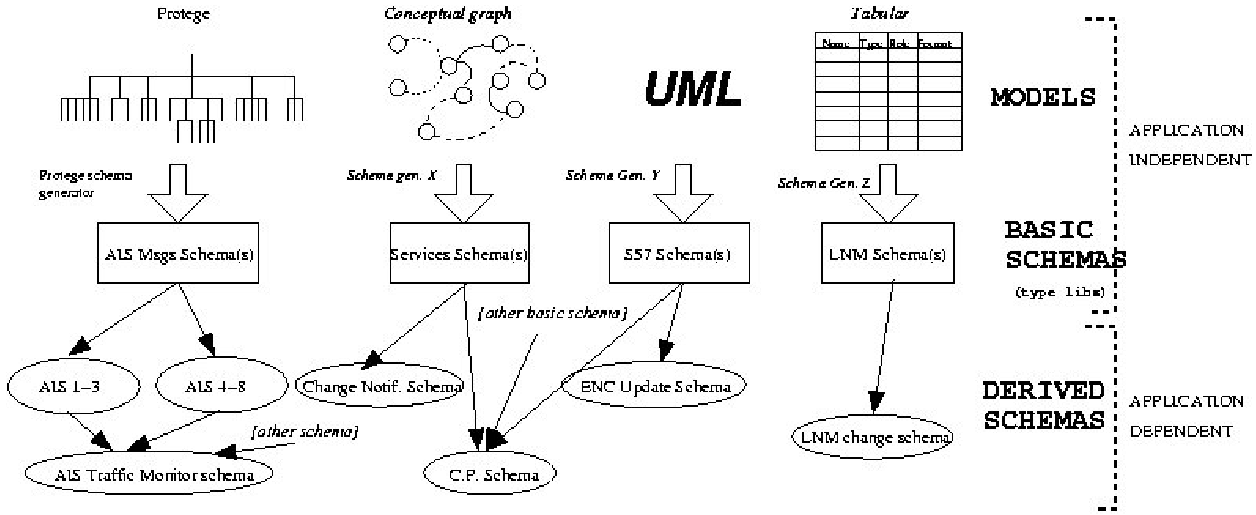

Figure 1: Relationship between models, type libraries, and application schemas

The major functional components of our approach are: (i) a r*egistry of models and schemas*, i.e., a "dictionary" of models, schemas, MIML tags, classes, attributes, metadata, etc.; (ii) *model and schema maintenance* functionality; and (iii) *model merging* capabilities for combining models. Sundry components, such as access control mechanisms and translators between different data formats, are also envisaged. Future incorporation of models in formats other than Protege and future use of RDF or OWL (Web Ontology Language) will be possible by adding appropriate convertors. Each model and type library will have its own namespace, while application schemas will have other namespaces. Questions of structural and naming conflicts, etc., will be addressed by limiting the scope of namespaces, defining special namespaces for widely-used data, and using the registry for preventing conflicts in the first place.

The advantages of this approach lie in the separation of domain descriptions from applications. The domains are described by the models, while the documents exchanged by applications are described by the application level schemas. This separation makes it easier for application designers to concentrate on data handling and information processing in their applications, without needing to expend a great deal of efforts understanding the domain, because the schemas – their representations of the domains – are guarantees of the form, relationships, and (to an extent) the content of the data they will need to handle. Errors in form and content can be detected through the XML processor and XML parsing libraries. Relationships between data elements are also available – it becomes possible for an application to know, for example, that latitude in decimal degrees and latitude in degrees+minutes are both representations of the same geolocational concept. The architecture also allows disparate models from disparate sources, expressed in different formats, to be included in WIN with a minimum of added effort. This addresses the logistic and technical problems involved in bringing in stakeholders from different sub-domains (e.g., meteorology, nautical charts, cargo service providers), each of whom may have significant effort invested in existing domain models for their own domains. Adding a new domain model now becomes a matter of writing a schema generator to convert that model format into a new type library .

Another important advantage is the retention of a clear relationship between XML tags and attributes, and entities and attributes in the underlying domains. Any tag or attribute belonging to a type library schema is unambiguously identified with one and only one class or attribute in one and only one model. This will be important for future intelligent applications which can reason about concept inter-relationships – for example, an intelligent information retrieval application which obtains information about weather from different sources will be able to figure out that sea conditions, temperature, and visibility are all different kinds of weather conditions. Future semantic web applications will need access to formal representations of MTS domain knowledge; this is provided for in our design.

4. Related Work

Ontolingua (Farquhar et al., 1997) provides an environment for publishing and editing ontologies and, in conjunction with Chimaera (McGuiness et al., 2000) provides ontology management features such as ontology merging and diagnostics. WebOnto/Tadzebao (Domingue et al 1998) is a collection of tools for ontology construction that allows geographically dispersed designers to work together. Functionality for distributed ontology creation functionality is planned for a future release of Protege (Grosso et al., 1999). There are also numerous efforts on database interoperability, schema merging, and ontology mapping – these differ from our solution in purpose and applicability, in that most address only model merging from the WIN perspective, and would apply only to specific provider-user information transfer channels (which are information transfer modes which WIN is supposed to replace).

6. Conclusion

This paper uses a community-based and decentralized approach to address, within the marine transportation domain, a smaller version of the similar problem faced by the Federal government concerning proliferating markup languages and the challenge of defining XML vocabularies (GAO 2002). It takes the suggested "bottom-up approach to establish a centralized registry of key XML data elements and structures and coordinate its use by XML systems developers" (GAO 2002) thereby giving MTS software developers an incentive to reuse vocabularies. It presents an architecture for bringing formal artificial intelligence concepts (computational ontologies) into application development. Since it gives application developers access to formalisms (e.g., computational ontologies and UML representations) that facilitate intelligent reasoning and information retrieval, it also lays a foundation for potential semantic web applications for the marine transportation system.

Acknowledgements

We would like to thank Pradnya Dharia, Prajakta Nivargi, Ernest Hunt, and Pete Smullen for their contributions. This work was supported by the U.S. Coast Guard, Sun Microsystems, and by the National Science Foundation under grant EIA-9983267. The opinions and recommendations in this paper are those of the authors and may not represent the views of the USCG or NSF.

References

(Domingue et al., 1998): J. Domingue: Tadzebao and WebOnto: discussing, browsing, and editing ontologies on the web. 11th Workshop on Knowledge Acquisition, Modeling and Management, Banff, Canada, 1998.

(Farquhar et al., 1997): A. Farquhar, R. Fikes, J. Rice: The Ontolingua server: a tool for collaborative ontology construction, International Journal of Human-Computer Studies, 46, 707-727, 1997.

(GAO 2002): General Accounting Office (GAO): Electronic government: challenges to effective adoption of the Extensible Markup Language, GAO Report 02-327, April 2002.

(Grosso et al., 1999): W. E. Grosso, H. Eriksson, R.W.Ferguson, J. H. Gennari, S. W. Tu, M. A. Musen: Knowledge modeling at the millennium: the design and evolution of Protege-2000. 12th Banff Workshop on Knowledge Acquisition, Modeling, and Management. Banff, Alberta, 1999.

(Malyankar 2002): R. M. Malyankar: Vocabulary development for markup languages – a case study with maritime information. Proc. 11th World Wide Web Conference, Honolulu, 2002, pp. 674-685.

(McGuiness et al. 2000): D. L. McGuinness, R. Fikes, J. Rice and S. Wilder: An environment for merging and testing large ontologies. Proc. 7th International Conference on Principles of Knowledge Representation and Reasoning, Breckenridge, Colorado, April 2000.

(NRC 1999): Committee on Maritime Advanced Info Systems, Marine Board, Commission on Engineering and Technical Systems: Applying advanced information systems to ports and waterways management, National

Academy Press (NAP), National Research Council, Washington DC, 1999.
(Spalding et al., 2002): J. W. Spalding, K. M. Shea, M. J. Lewandowski: Intelligent waterway system and the Waterway Information Network. Institute of Navigation National Technical Meeting, San Diego, January 2002.
(USCG-PMG 2001) U. S. Coast Guard Office of Waterways Management (G-WM)/Potomac Management Group (PMG): An assessment of the Integrated Maritime Information System (IMIS) concept as applied to U.S. ports and waterways, February 2001.
(USDOT 1999): U. S. Department of Transportation (DOT): An assessment of the U. S. marine transportation system: a report to Congress, September 1999.